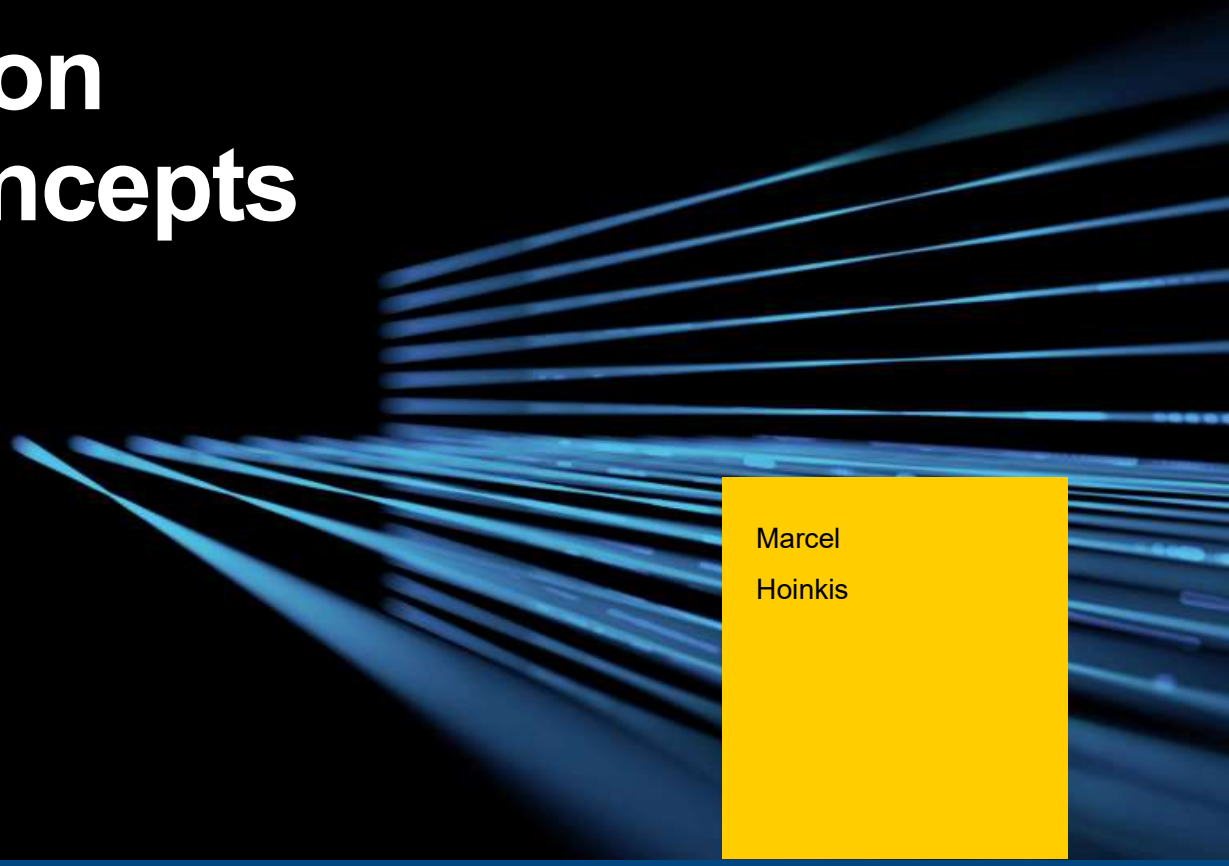
Sulzer

hroug 23 annual conference

Practical Application of historization concepts



Marcel Hoinkis

Sulzer

hroug annual conference

APEX @ Sulzer

Personal Info

Marcel Hoinkis

Project Manger
APEX Consultant

at Sulzer GmbH in Munich

Sulzer

hroug 23 annual conference

Questions to answer

Why would you need historization?

Which implementation should you use? Why?

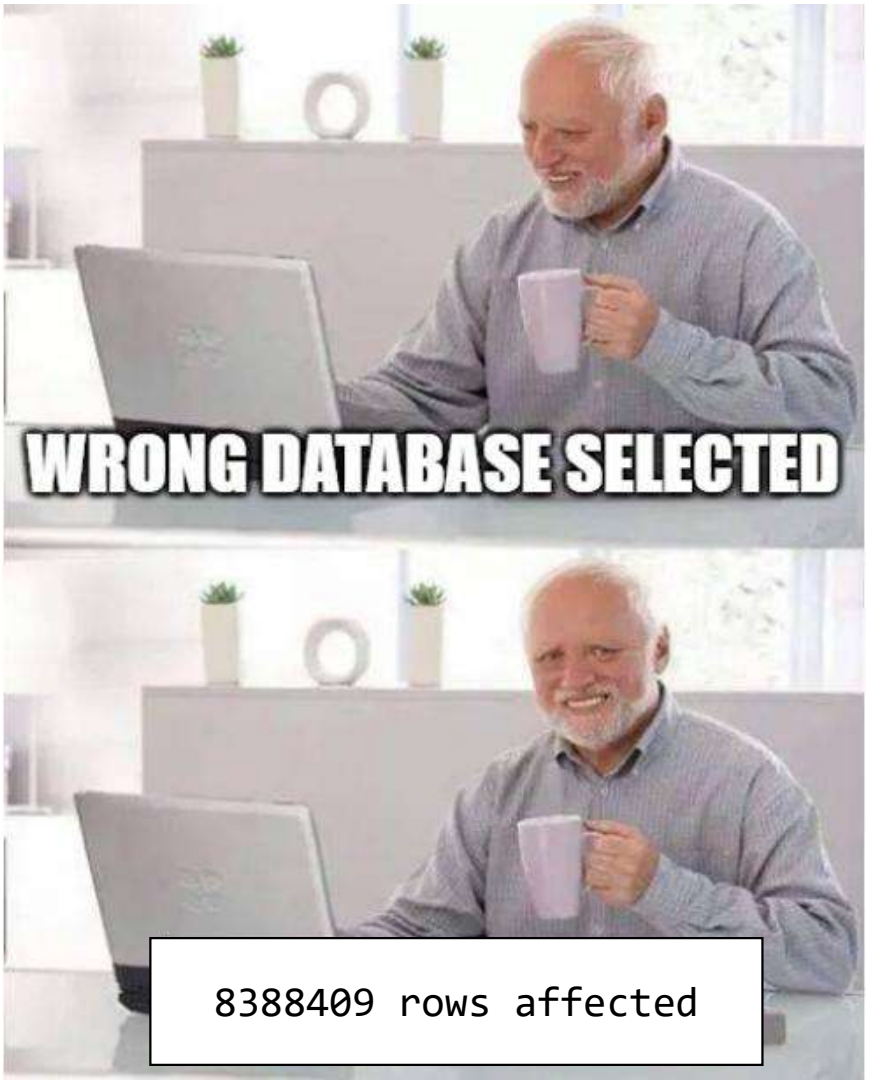
What do you need to enable it?

Sulzer

hroug'23 annual conference

The Historization Definition today

- Save past data in our tables
- Have direct access to this data
- Insuring “no gaps”
- Traceability of changes


WRONG DATABASE SELECTED
8388409 rows affected

Sulzer

hroug 23 annual conference

Why though? – some scenarios

⟶ Users change important process data by accident

⟶ connected system dumps garbage data that users already worked with

⟶ Angry user goes through your data like tornado

⟶ Your colleague forgot the where clause on his delete action again

Legal/Compliance reasons

Required by the business process

Rules you set as dev

Sulzer

hroug 23 annual conference

The usual implementation

```
create or replace trigger trg_trigger_test
after insert or update or delete on transaction_test_trigger
for each row
declare
    l_operation varchar2(1) := case when updating then 'U' when deleting then 'D' else 'I' end;
begin
    if updating or inserting then
        insert into transaction_test_trigger_hist(t_id, num_col, char_col, date_col, operation)
        values(:new.t_id, :new.num_col, :new.char_col, :new.date_col, l_operation);
    else
        insert into transaction_test_trigger_hist(t_id, num_col, char_col, date_col, operation)
        values(:old.t_id, :old.num_col, :old.char_col, :old.date_col, l_operation);
    end if;
end;
/
```

Differentiation and comparison of historization methods

Triggers

Table API

Flashback

Time Travel

(„Total Recall“, Flashback Data Archive FDA)

Differentiation and comparison of historization methods

Triggers

+ No “nieche” knowledge needed for setup
+ No additional rights needed
+ No hassle with your DBA

- Nightmare to keep up to date
- Poor Performance
- Developing queries is complex and lots of requires thinking
- Retention needs to be handled separately
- Not tamper proof

Sulzer

hroug 23 annual conference

Differentiation and comparison of historization methods

Table API

+ Easy to implement
+ Fits for already existing Table APIs
+ No hassle with your DBA

(Package that handles CRUD operations for you:
Package with different procedures
Is dependend on the structure of the table)

- Nightmare to keep up to date
- Poor Performance
- Developing queries is complex and lots of requires thinking
- Retention needs to be handled separately
- Not tamper proof

Sulzer

hroug 23 annual conference

Differentiation and comparison of historization methods

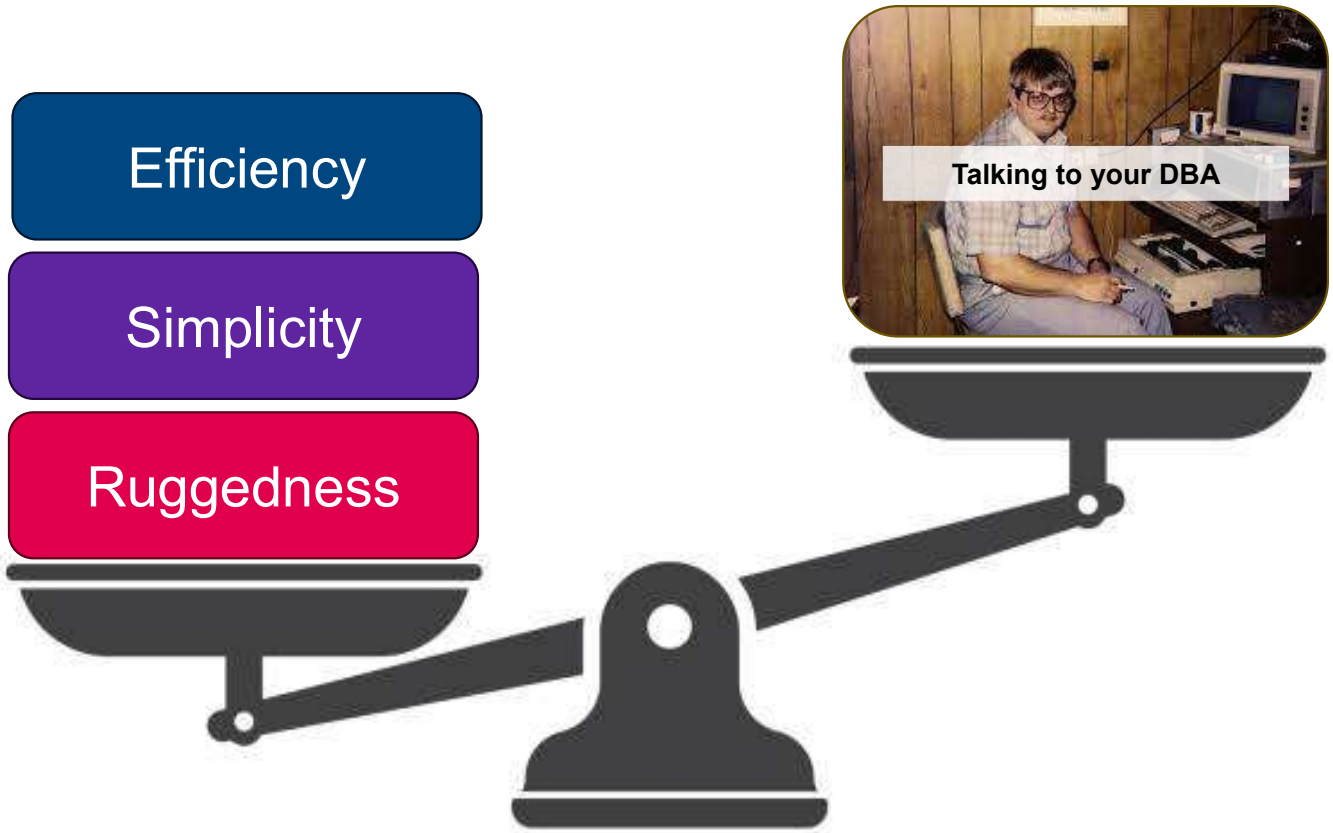
Flashback Time Travel

+ Performance is significantly better
+ Keeps itself up to date with your DM
+ Easiest solution to implement
+ Retention handled by DB
+ Tamper proof

- Additional permissions needed
- Seems to be not that well known
- History Table names can be tricky

Differentiation and comparison of historization methods

Summary:



Sulzer

hroug 23 annual conference

Basic performance test

Efficiency

Transaction Metric	Base Updates	Time Travel	Trigger based
Number of Rows	2.000.000	2.000.000	2.000.000
Duration in s	15,29	18,29	144,06
Rows/s	130834	109367	13883
Performance Impact	100%	120%	960%

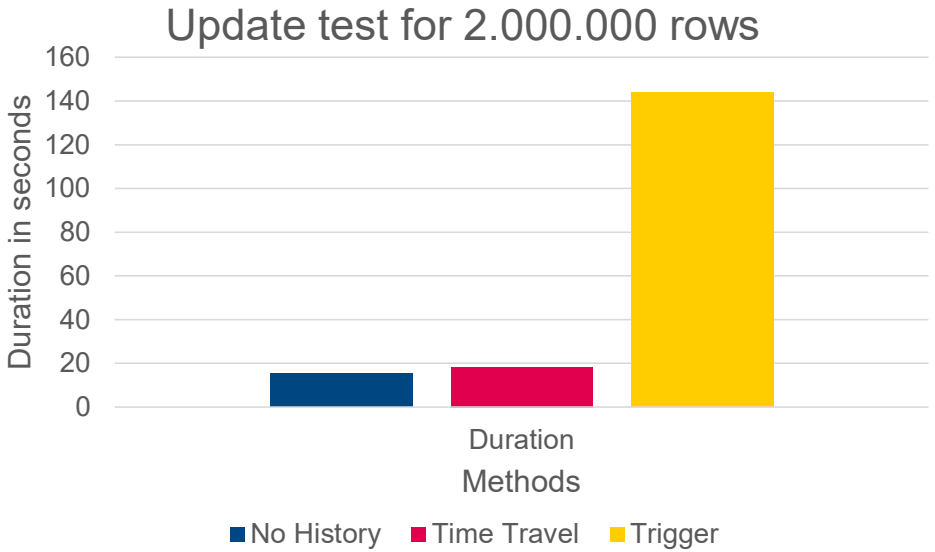
Sulzer

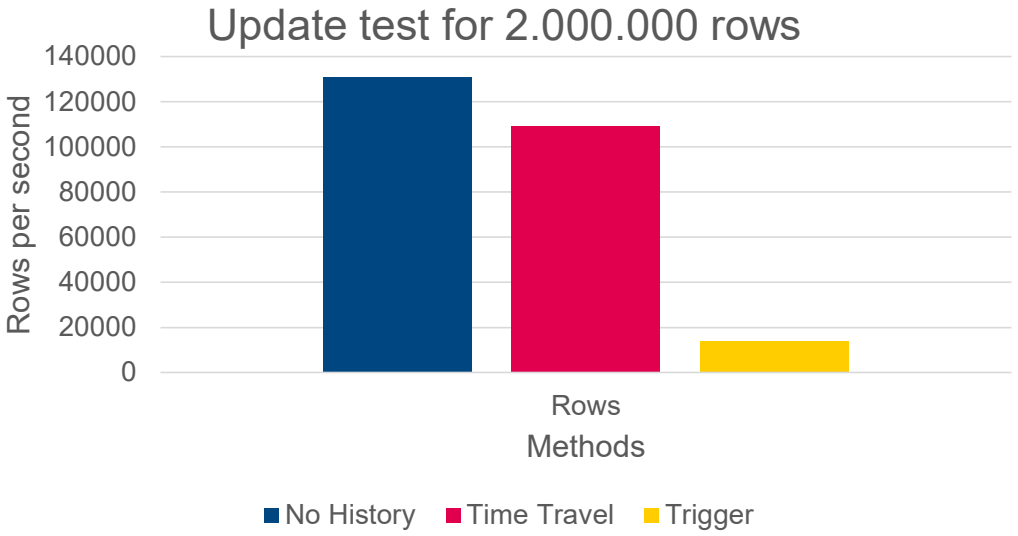
hroug 23 annual conference

Basic performance test

Efficiency

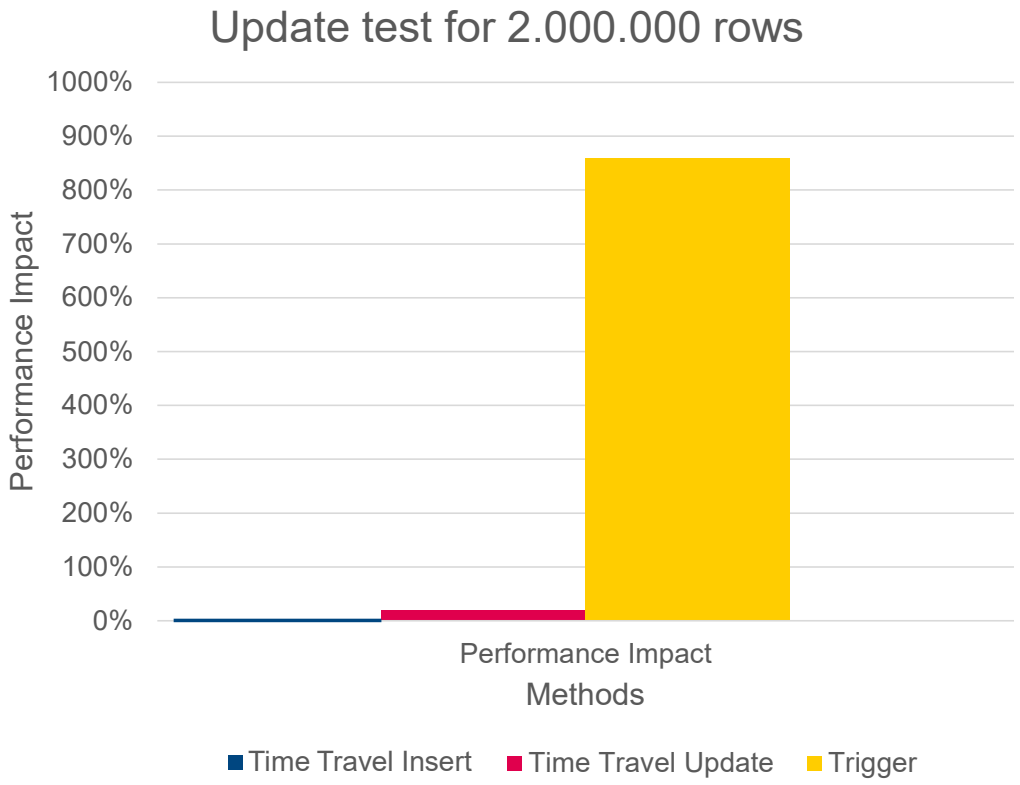
Transaction Metric	Base Updates	Time Travel	Trigger based
Number of Rows	2.000.000	2.000.000	2.000.000
Duration in s	15,29	18,29	144,06
Rows/s	130834	109367	13883
Performance Impact	100%	120%	960%





Basic performance summary

Efficiency



marginal Performance Impact with TT
big Performance Impact with triggers

Sulzer

hroug 23 annual conference

The magic in Time Travel

Sulzer

hroug 23 annual conference

How Time Travel works

Simplicity

```
create flashback archive archive_hr_test tablespace users quota 5G retention 5 day;

alter table employees flashback archive archive_hr_test;
```

Sulzer

hroug 23 annual conference

How Time Travel works

Simplicity

- `SELECT * FROM employees AS OF TIMESTAMP systimestamp - 14`

`WHERE employee_id = 100`

Present data:

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID
100	Steven	King	SKING	515.123.4567	17.06.03 00:00:00	AD_PRES	24869,6	(null)	(null)	90

Past data: (as of)

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID
100	Steven	King	SKING	515.123.4567	17.06.03 00:00:00	AD_PRES	24000	(null)	(null)	90

Can be as of timestamp, scn

Specifies a point in time you want to see the data at

Sulzer

hroug 23 annual conference

How Time Travel works

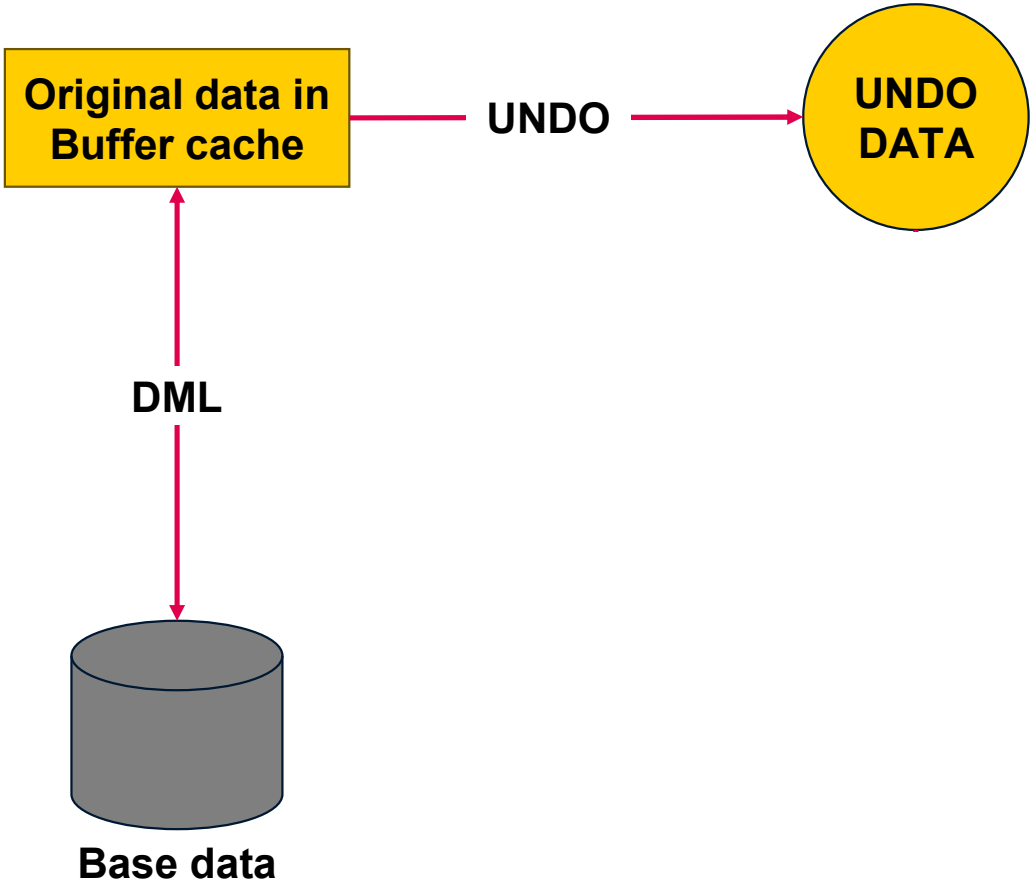
Simplicity

```
select
    emp.*,
    versions_starttime,
    versions_endtime,
    versions_xid,
    versions_operation,
    DBMS_FLASHBACK_ARCHIVE.get_sys_context(versions_xid, 'USERENV','SESSION_USER') AS session_user,
    DBMS_FLASHBACK_ARCHIVE.get_sys_context(versions_xid, 'USERENV','CLIENT_IDENTIFIER') AS client_identifier
from employees versions between scn minvalue and maxvalue emp
where emp.employee_id = 100;
```

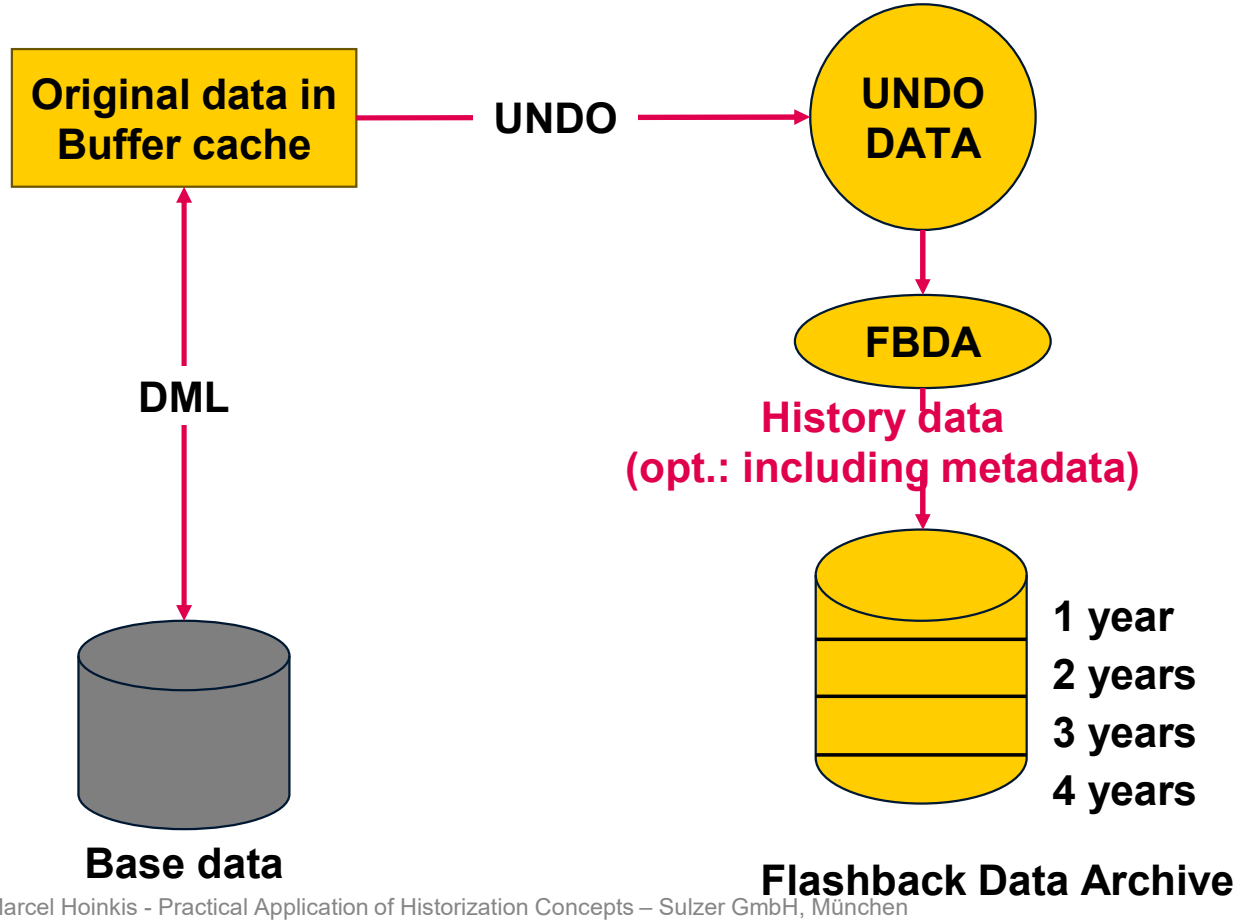
	LAST_NAME	SALARY	VERSIONS_STARTTIME	VERSIONS_ENDTIME	VERSIONS_OPERATION	SESSION_USER	CLIENT_IDENTIFIER
1	Smith	25113,32	13.10.23 12:18:31...	13.10.23 12:21:25...	I	HR	David Dürer
2	Not Smith	25113,32	13.10.23 12:21:25...	13.10.23 12:23:40...	U	HR	Patrizia Regenberg
3	Not Smith	25615,59	13.10.23 12:23:40...	(null)	U	HR	Marcel Hoinkis

Shows the versions between two scns or timestamps

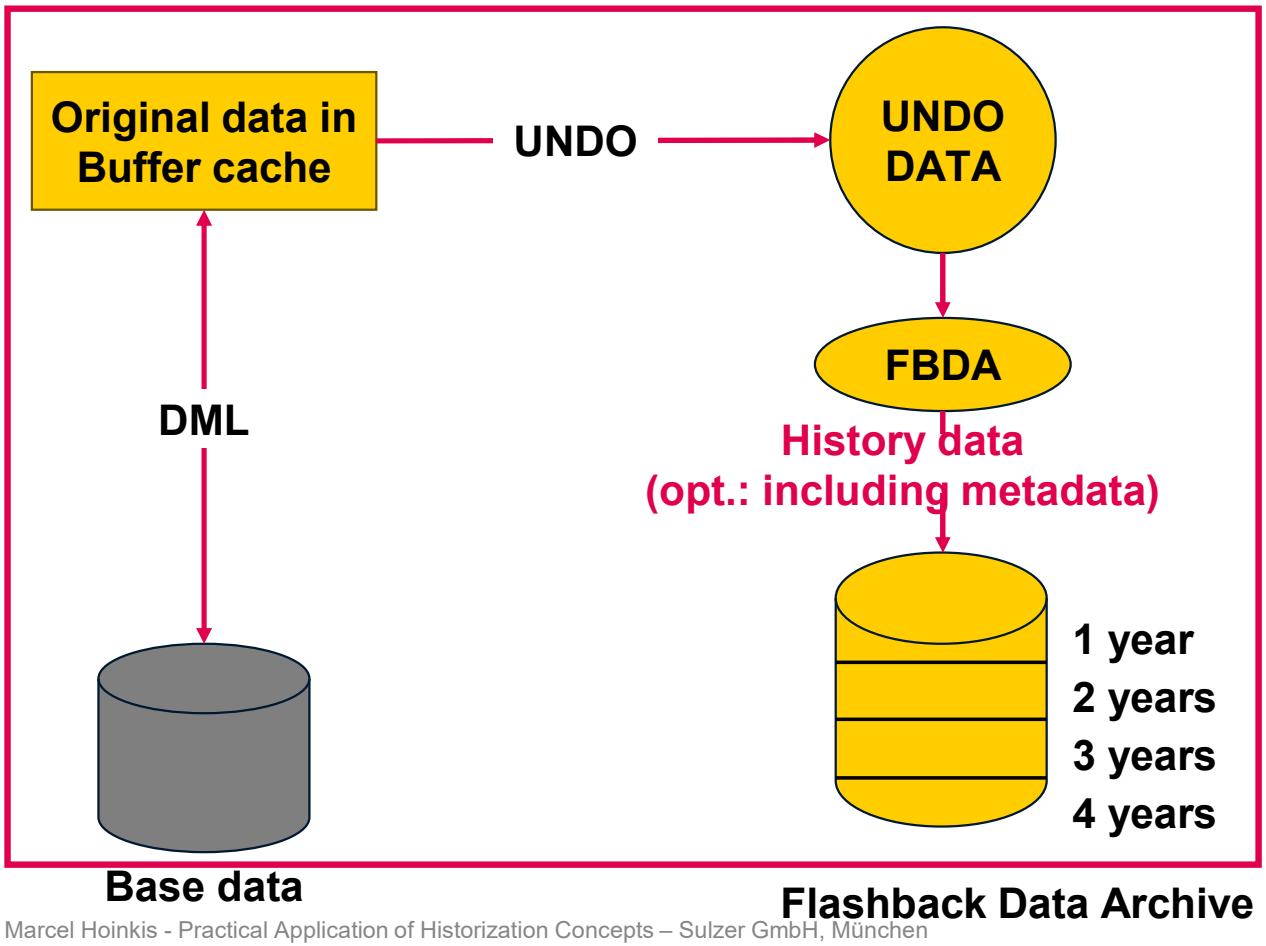
How does Oracle Flashback Time Travel work?



How does Oracle Flashback Time Travel work?



How does Oracle Flashback Time Travel work?



Ruggedness

Efficiency

Simplicity

Source: Satishbabu G., 14.08.2023
https://www.linkedin.com/pulse/mastering-data-history-guide-using-oracles-flashback-archive

Sulzer

hroug 23 annual conference

Too good to be true?

This costs extra, doesn't it?

Free since DB Release 12c*

Prerequisites of Oracle Flashback Time Travel

Your duties

Be logged in as SYSDBA or have FLASHBACK ARCHIVE ADMINISTER privilege for your schema

- CREATE FLASHBACK ARCHIVE
- ALTER FLASHBACK ARCHIVE
- DROP FLASHBACK ARCHIVE)

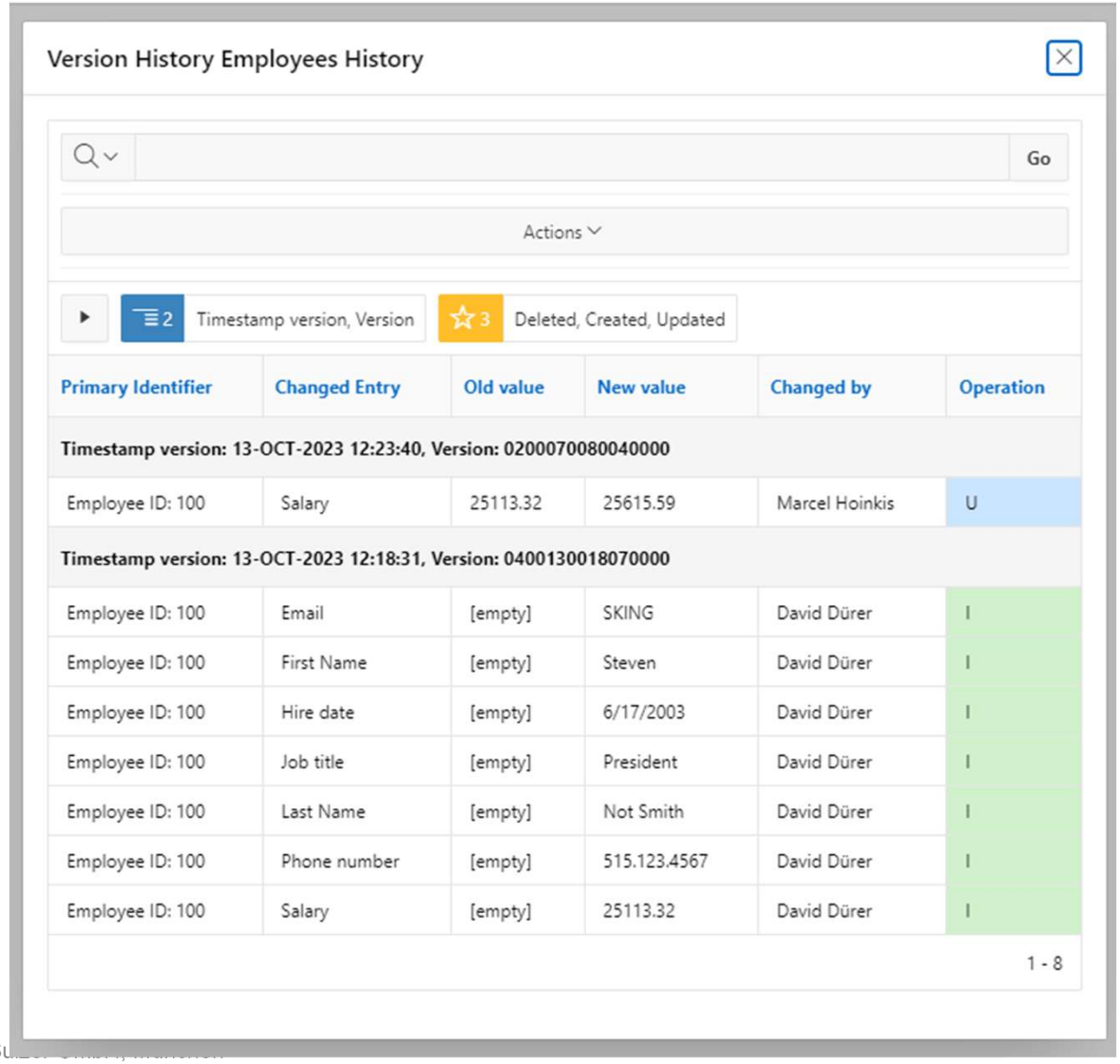
Tell your DBA your specific needs

- Retention
- Sys context,
- quotas needed,
- additional tablespace(s),
- compression??!

Sulzer

hroug 23 annual conference

Examples of Integrating it into your application's UI



Sulzer

```
select
    'EMPLOYEES' as tracked_table,
    employee_id as primary_key,
    employee_id as object_key,
    'Employee ID: ' || employee_id as primary_identifier,
    changed_column,
    col_value,
    versions_starttime,
    versions_endtime,
    versions_xid,
    versions_operation
from (
    select
        emp.attribute
        j.job_title,
        versions_starttime,
        versions_endtime,
        versions_xid,
        versions_operation
    from employees versions between scn minvalue and maxvalue emp
    left join jobs j on j.job_id = emp.job_id
)
unpivot (
    col_value
    for changed_column in (
        first_name          as 'First Name',
        last_name           as 'Last Name',
        email               as 'Email',
        phone_number        as 'Phone number',
        hire_date           as 'Hire date',
        salary              as 'Salary',
        job_title           as 'Job title'
    )
);
```

Standardized form for all views 3

Any attributes from history table

All versions of the tracked table 1

Versions attributes get added

Multiple Tables can be joined (lookup values)

Required Attributes get unpivoted

→One entry for every tracked column 2

→Changed Attribute is shown

TRACKED_TABLE	PRIMARY_KEY	OBJECT_KEY	PRIMARY_IDENTIFIER	CHANGED_COLUMN	COL_VALUE	VERSIONS_STARTTIME	VERSIONS_ENDTIME	VERSIONS_XID	VERSIONS_OPERATION
EMPLOYEES	100	100	Employee ID: 100	First Name	Steven	13.10.23 12:18:31,0000000	13.10.23 12:21:25,00000(	0400130018070000	I
EMPLOYEES	100	100	Employee ID: 100	Last Name	Not Smith	13.10.23 12:18:31,0000000	13.10.23 12:21:25,00000(	0400130018070000	I
EMPLOYEES	100	100	Employee ID: 100	Email	SKING	13.10.23 12:18:31,0000000	13.10.23 12:21:25,00000(	0400130018070000	I
EMPLOYEES	100	100	Employee ID: 100	Phone number	515.123.4567	13.10.23 12:18:31,0000000	13.10.23 12:21:25,00000(	0400130018070000	I
EMPLOYEES	100	100	Employee ID: 100	Hire date	17.06.03 00:00	13.10.23 12:18:31,0000000	13.10.23 12:21:25,00000(	0400130018070000	I
EMPLOYEES	100	100	Employee ID: 100	Salary	25113,32	13.10.23 12:18:31,0000000	13.10.23 12:21:25,00000(	0400130018070000	I
EMPLOYEES	100	100	Employee ID: 100	Job title	President	13.10.23 12:18:31,0000000	13.10.23 12:21:25,00000(	0400130018070000	I
EMPLOYEES	100	100	Employee ID: 100	First Name	Steven	13.10.23 12:21:25,0000000	13.10.23 12:23:40,00000(	01000A00FC05000	U
EMPLOYEES	100	100	Employee ID: 100	Last Name	Not Smith	13.10.23 12:21:25,0000000	13.10.23 12:23:40,00000(	01000A00FC05000	U
EMPLOYEES	100	100	Employee ID: 100	Email	SKING	13.10.23 12:21:25,0000000	13.10.23 12:23:40,00000(	01000A00FC05000	U
EMPLOYEES	100	100	Employee ID: 100	Phone number	515.123.4567	13.10.23 12:21:25,0000000	13.10.23 12:23:40,00000(	01000A00FC05000	U
EMPLOYEES	100	100	Employee ID: 100	Hire date	17.06.03 00:00	13.10.23 12:21:25,0000000	13.10.23 12:23:40,00000(	01000A00FC05000	U
EMPLOYEES	100	100	Employee ID: 100	Salary	25113,32	13.10.23 12:21:25,0000000	13.10.23 12:23:40,00000(	01000A00FC05000	U
EMPLOYEES	100	100	Employee ID: 100	Job title	President	13.10.23 12:21:25,0000000	13.10.23 12:23:40,00000(	01000A00FC05000	U
EMPLOYEES	100	100	Employee ID: 100	First Name	Steven	13.10.23 12:23:40,000000000		0200070080040000	U
EMPLOYEES	100	100	Employee ID: 100	Last Name	Not Smith	13.10.23 12:23:40,000000000		0200070080040000	U
EMPLOYEES	100	100	Employee ID: 100	Email	SKING	13.10.23 12:23:40,000000000		0200070080040000	U
EMPLOYEES	100	100	Employee ID: 100	Phone number	515.123.4567	13.10.23 12:23:40,000000000		0200070080040000	U
EMPLOYEES	100	100	Employee ID: 100	Hire date	17.06.03 00:00	13.10.23 12:23:40,000000000		0200070080040000	U
EMPLOYEES	100	100	Employee ID: 100	Salary	25615,59	13.10.23 12:23:40,000000000		0200070080040000	U
EMPLOYEES	100	100	Employee ID: 100	Job title	President	13.10.23 12:23:40,000000000		0200070080040000	U

Sulzer

hroug 23 annual conference

```
create or replace function table_change_history_columnar(history_view dbms_tf.table_t, object_key_filter varchar2 default null) return varchar2 sql_macro
is
begin
    return q'{
        select
            hd.tracked_table,
            hd.primary_key,
            hd.object_key,
            hd.primary_identifier,
            hd.changed_column,
            nvl(hd.old_col_value, '[empty]') as old_col_value,
            nvl(hd.new_col_value, '[empty]') as new_col_value,
            hd.changed_by,
            hd.versions_starttime,
            hd.versions_xid,
            hd.versions_operation
        from (
            select *
            from (
                select
                    tracked_table,
                    primary_key,
                    object_key,
                    primary_identifier,
                    changed_column,
                    changed_by,
                    versions_starttime,
                    versions_xid,
                    versions_operation,
                    lag(col_value, 1) over(partition by tracked_table, primary_key order by versions_starttime) as old_col_value,
                    col_value as new_col_value
                from (
                    select
                        tracked_table,
                        primary_key,
                        object_key,
                        primary_identifier,
                        changed_column,
                        case when versions_operation = 'D' then null else col_value end as col_value,
                        nvl(case when versions_xid is not null then dbms_flashback_archive.get_sys_context(versions_xid, 'USERENV','CLIENT_IDENTIFIER') end, 'unknown') as changed_by,
                        nvl(versions_starttime, to_timestamp('01.01.1900', 'DD.MM.YYYY')) as versions_starttime,
                        nvl(versions_xid, '0') as versions_xid,
                        nvl(versions_operation, 'I') as versions_operation
                    from history_view
                    where object_key = table_change_history_columnar.object_key_filter
                    or table_change_history_columnar.object_key_filter is null
                ) bhd
            )
            where old_col_value != new_col_value
            or (old_col_value is null and new_col_value is not null)
            or (new_col_value is null and old_col_value is not null)
        ) hd
    }';
end table_change_history_columnar;
/
```

1 SQL Macro

Valid Data Source as Input

Returns a standardized form for any historized table includes audit information 5

Window function gets old and new value 3

Selects standardized attributes from view

Retrieves sys context 2

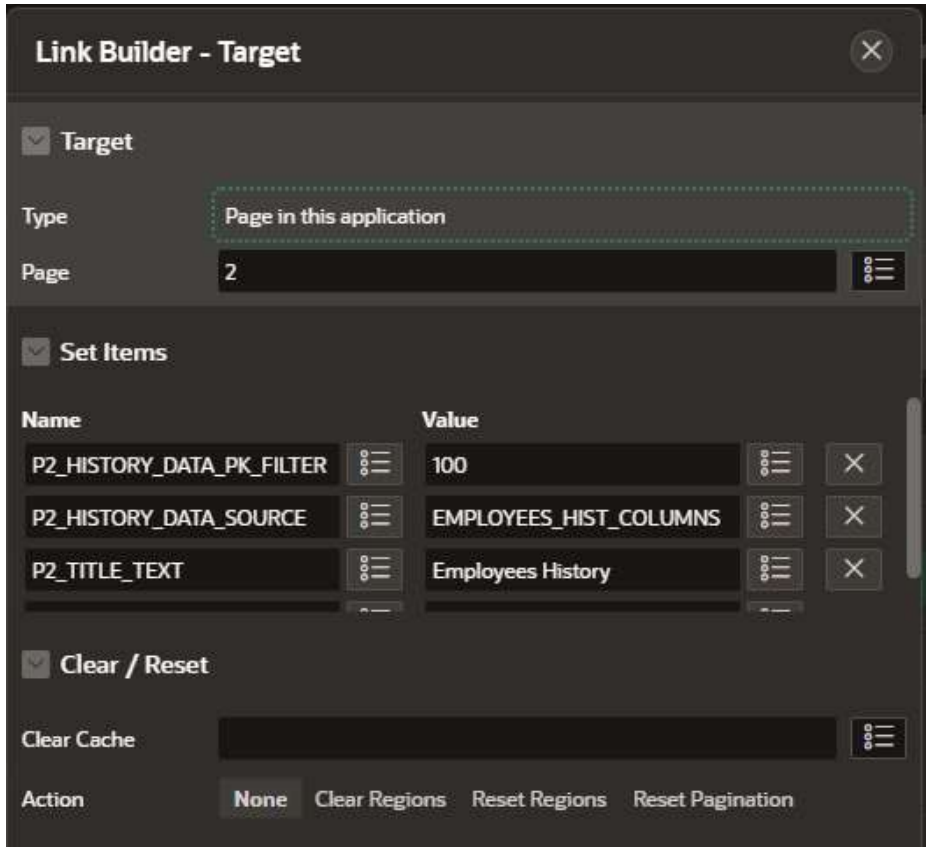
Only where old is not new value 4

TRACKED_TABLE	PRIMARY_KEY	OBJECT_KEY	PRIMARY_IDENTIFIER	CHANGED_COLUMN	OLD_COL_VALUE	NEW_COL_VALUE	CHANGED_BY	VERSIONS_STARTTIME	VERSIONS_XID	VERSIONS_OPERATION
EMPLOYEES	100	100	Employee ID: 100	Email	[empty]	SKING	David Dürer	13.10.23 12:18:31,0000000	0400130018070000	I
EMPLOYEES	100	100	Employee ID: 100	First Name	[empty]	Steven	David Dürer	13.10.23 12:18:31,0000000	0400130018070000	I
EMPLOYEES	100	100	Employee ID: 100	Hire date	[empty]	17.06.03 00:00:00	David Dürer	13.10.23 12:18:31,0000000	0400130018070000	I
EMPLOYEES	100	100	Employee ID: 100	Job title	[empty]	President	David Dürer	13.10.23 12:18:31,0000000	0400130018070000	I
EMPLOYEES	100	100	Employee ID: 100	Last Name	[empty]	Not Smith	David Dürer	13.10.23 12:18:31,0000000	0400130018070000	I
EMPLOYEES	100	100	Employee ID: 100	Phone number	[empty]	515.123.4567	David Dürer	13.10.23 12:18:31,0000000	0400130018070000	I
EMPLOYEES	100	100	Employee ID: 100	Salary	[empty]	25113,32	David Dürer	13.10.23 12:18:31,0000000	0400130018070000	I
EMPLOYEES	100	100	Employee ID: 100	Salary	25113,32	25615,59	Marcel Hoinkis	13.10.23 12:23:40,0000000	0200070080040000	U

Examples of Integrating it into your application's UI

- Create a structured view that selects the historic values and unpivots them
- Optional: union multiple views to combine different table histories
  - More than one historized table can be shown
- Create a SQL Macro that receives the view name
- Create 1 page that shows the results of the macro (Input is any view)

➡ UI that returns changed values for any business object



Sulzer

Any questions?

Wishing you a great hroug 2023!